

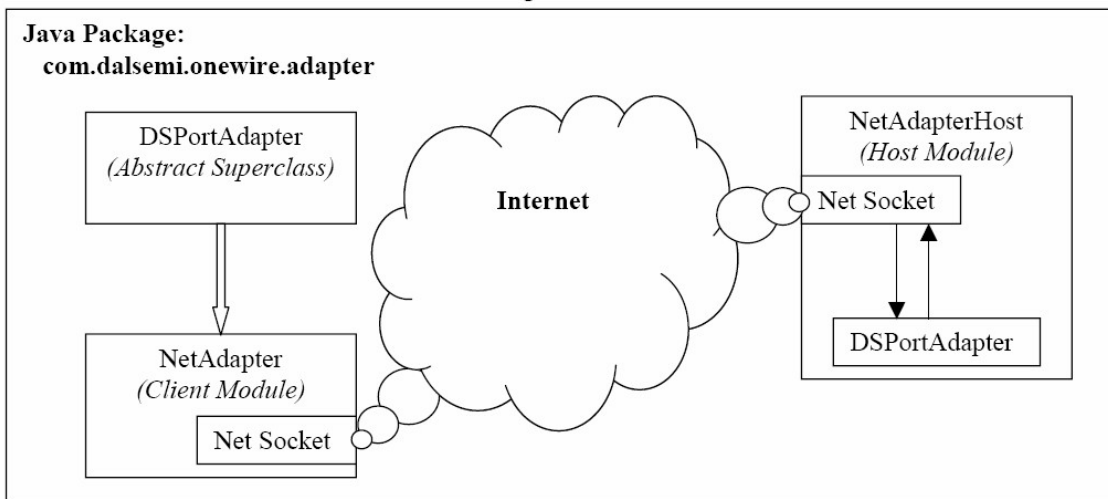
# 1-Wire Net Adapter

Consider this scenario: You have a small shed just outside your house. Your weather station is attached to its roof. In the shed you have your old trusty 700 MHz Pentium 3 running your weather station, and it's connected to your house network with WiFi. You want to make some changes to your weather station software, but unfortunately it's hot outside and your shed doesn't have AC. One solution is to extend your 1-Wire into your house and use your main computer, but this doesn't seem right.

Or try this: Maybe you live in an area susceptible to lightning and you'd like to have your weather station computer as far from your main computer and electronics as possible... like in the garage. You've seen old PCs or Macs on eBay for just a few dollars, and these days you can get an 802.11g wireless card for less than \$20. Hmmmm....

I have my old computer running my weather station and thought it would be neat to write some code that would allow me to access my 1-Wire network across a standard wired or wireless home network. When I wanted to work on the software, I could quit the weather software and run some sort of a network adapter program. It would act like the 1-Wire bus is connected right to my computer. Once the code was debugged, I could copy to the old computer.

As I started to write some code, I noticed that buried in the 1-Wire Java API was something called a Net Adapter. Could this be what I was going to write? A little digging and, sure enough, DalSemi/Maxim has already done this! Using a networked computer or a Tini, you can access your 1-Wire bus thru a standard Ethernet network, anywhere on the Internet! This opens the door to many possibilities.



Net Adapter Topology and Class Diagram from Maxim App Note 193

# 1-Wire Net Adapter

The code for this project is pretty much lifted right from the example code provided in the Java OneWireAPI download. I converted it to NetBeans and added a simple way to quit the program. To use it, there are just 2 simple steps:

- 1) On the PC or Mac your 1-Wire bus is connected to, launch NetBeans and run the NetAdapter code. If you're using Tini, there's a Tini version in the OneWireAPI examples folder (you may have to rebuild it to run on your version of TiniOS). This computer is the host. You'll also need to know the IP Address of this machine for step 2.
- 2) On your development (or main) computer, called the client, change the adapter type to NetAdapter and the adapter name (or com port name) to the IP address of the host. In SimpleWeather, it would look like this:

```
// user constants
public static final String VERSION = "SimpleWeather 1.0";
public static final String ONE_WIRE_SERIAL_PORT = "192.168.1.45";

...

// class constants
public static final String ADAPTER_TYPE = "NetAdapter";

...
```

If you're using TWS version 5, change the prefs.ini file:

```
# Primary 1-Wire Adapter
Adapter_Type           = NetAdapter
Adapter_Com_Port      = 147.24.152.106
```

When you run the client, it will look on the network for the machine specified in the IP Address, and then look for its 1-Wire Adapter. If it finds it, it will run just as if the 1-Wire network is connected directly to the client machine. Pretty neat, huh?

DalSemi/Maxim has put together a Application Note on this very topic. They call it a Network Proxy in app Note 193. Check it out at [http://www.maxim-ic.com/appnotes.cfm/an\\_pk/193](http://www.maxim-ic.com/appnotes.cfm/an_pk/193)

Since the two computers are connected over a standard network, you could in theory have your weather station anywhere: outside in the shed or half-way around the world. All you need is an Internet connection and an IP address. Since the main software is running at your location, you have complete control. For example, suppose you have temperature sensor at your business monitoring a freezer (the host). If the freezer fails, you could turn on a warning light or a buzzer at your home (the client, using SimpleWeather). Try doing that with a store-bought weather station!